UNIVERSAL PROGRAMMABLE PERIPHERAL INTERFACE — 8-BIT MICROCOMPUTER

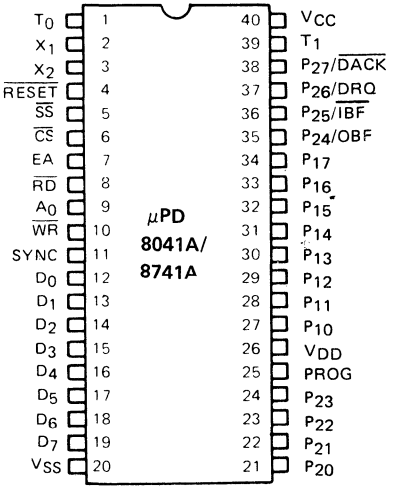
DESCRIPTION

The μPD8041A/8741A is a programmable peripheral interface intended for use in a wide range of microprocessor systems. Functioning as a totally self-sufficient controller, the μPD8041A/8741A contains an 8-bit CPU, 1K x 8 program memory, 64 x 8 data memory, I/O lines, counter/timer, and clock generator in a 40-pin DIP. The bus structure, data registers, and status register enable easy interface to 8048, 8080A or 8085A based systems. The μPD8041A's program memory is factory mask programmed, while the μPD8741A's program memory is UV EPROM to enable user flexibility.

FEATURES

- Fully Compatible with 8048, 8080A, 8085A and 8086 Bus Structure
- 8-Bit CPU with 1K x 8 ROM, 64 x 8 RAM, 8-Bit Timer/Counter, 18 I/O Lines
- 8-Bit Status and Two Data Registers for Asynchronous Slave-to-Master Interface
- Interchangeable EPROM and ROM Versions
- Interrupt, DMA or Polled Operation
- Expandable I/O
- 40-Pin Plastic or Ceramic Dip
- Single +5V Supply

PIN CONFIGURATION



7

PIN IDENTIFICATION

PIN NO.	PIN SYMBOL	FUNCTION
1,39	T_0, T_1	Testable input pins using conditional transfer functions JT0, JNT0, JT1, JNT1. T_1 can be made the counter/timer input using the STRT CNT instruction. The PROM programming and verification on the μPD8741A uses T_0.
2	X_1	One side of the crystal input for external oscillator or frequency source.
3	X_2	The other side of the crystal input.
4	$\overline{RESET}$	Active-low input for processor initialization. $\overline{RESET}$ is also used for PROM programming, verification, and power down.
5	$\overline{SS}$	Single Step input (active-low). $\overline{SS}$ together with SYNC output allows the μPD8741A to "single-step" through each instruction in program memory.
6	$\overline{CS}$	Chip Select input (active-low). $\overline{CS}$ is used to select the appropriate μPD8041A/8741A on a common data bus.
7	EA	External Access input (active-high). A logic "1" at this input commands the μPD8041A/8741A to perform all program memory fetches from external memory.
8	$\overline{RD}$	Read strobe input (active-low). $\overline{RD}$ will pulse low when the master processor reads data and status words from the DATA BUS BUFFER or Status Register.
9	A_0	Address input which the master processor uses to indicate if a byte transfer is a command or data.
10	$\overline{WR}$	Write strobe input (active-low). $\overline{WR}$ will pulse low when the master processor writes data or status words to the DATA BUS BUFFER or Status Register.
11	SYNC	The SYNC output pulses once for each μPD8041A/8741A instruction cycle. It can function as a strobe for external circuitry. SYNC can also be used together with $\overline{SS}$ to "single-step" through each instruction in program memory.
12-19	D_0-D_7 BUS	The 8-bit, bi-directional, tri-state DATA BUS BUFFER lines by which the μPD8041A/8741A interfaces to the 8-bit master system data bus.
20	V_{SS}	Processor's ground potential.
21-24, 35-38	P_{20}-P_{27}	PORT 2 is the second of two 8-bit, quasi-bi-directional I/O ports. P_{20}-P_{23} contain the four most significant bits of the program counter during external memory fetches. P_{20}-P_{23} also serve as a 4-bit I/O bus for the μPD8243, INPUT/OUTPUT EXPANDER. P_{24}-P_{27} can be used as port lines or can provide Interrupt Request (IBF and OBF) and DMA handshake lines (DRG and DACK).
25	PROG	Program Pulse. PROG is used in programming the μPD8741A. It is also used as an output strobe for the μPD8243.
26	V_{DD}	V_{DD} is the programming supply voltage for programming the μPD8741A. It is +5V for normal operation of the μPD8041A/8741A. V_{DD} is also the Low Power Standby input for the ROM version.
27-34	P_{10}-P_{17}	PORT 1 is the first of two 8-bit quasi-bi-directional I/O ports.
40	V_{CC}	Primary power supply. V_{CC} must be +5V for programming and operation of the μPD8741A and for the operation of the μPD8041A.

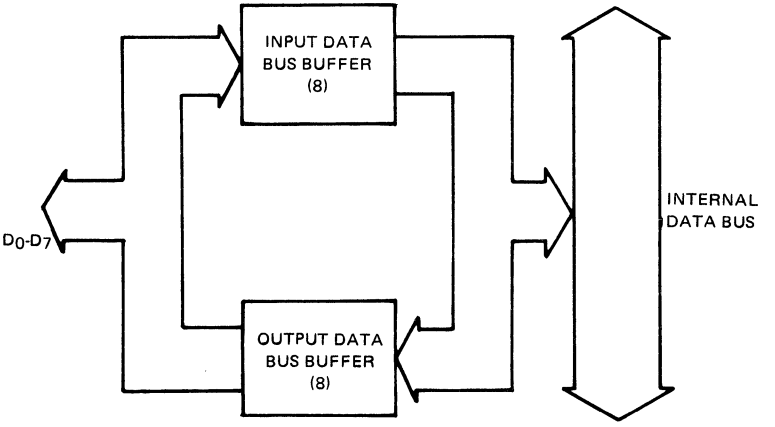
FUNCTIONAL DESCRIPTION

The μPD8041A/8741A is a programmable peripheral controller intended for use in master/slave configurations with 8048, 8080A, 8085A, 8086 — as well as most other 8-bit and 16-bit microprocessors. The μPD8041A/8741A functions as a totally self-sufficient controller with its own program and data memory to effectively unburden the master CPU from I/O handling and peripheral control functions. The μPD8041A/8741A is an intelligent peripheral device which connects directly to the master processor bus to perform control tasks which off load main system processing and more efficiently distribute processing functions.

μPD8041A/8741A FUNCTIONAL ENHANCEMENTS

The μPD8041A/8741A features several functional enhancements to the earlier μPD8041 part. These enhancements enable easier master/slave interface and increased functionality.

1. Two Data Bus Buffers. Separate Input and Output data bus buffers have been provided to enable smoother data flow to and from master processors.

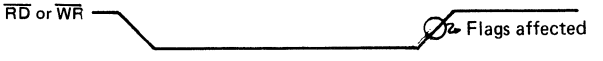


2. 8-Bit Status Register. Four user-definable status bits, ST_4-ST_7, have been added to the status register. ST_4-ST_7 bits are defined with the MOV STS, A instruction which moves accumulator bits 4-7 to bits 4-7 of the status register. ST_0-ST_3 bits are not affected.

ST_7	ST_6	ST_5	ST_4	F_1	F_0	IBF	OBF
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

MOV STS, A Instruction OP Code 90H

3. $\overline{RD}$ and $\overline{WR}$ inputs are edge-sensitive. Status bits IBF, OBF, F1 and INT are affected on the trailing edge at $\overline{RD}$ or $\overline{WR}$.



7

μPD8041A/8741A
FUNCTIONAL
ENHANCEMENTS (CONT.)

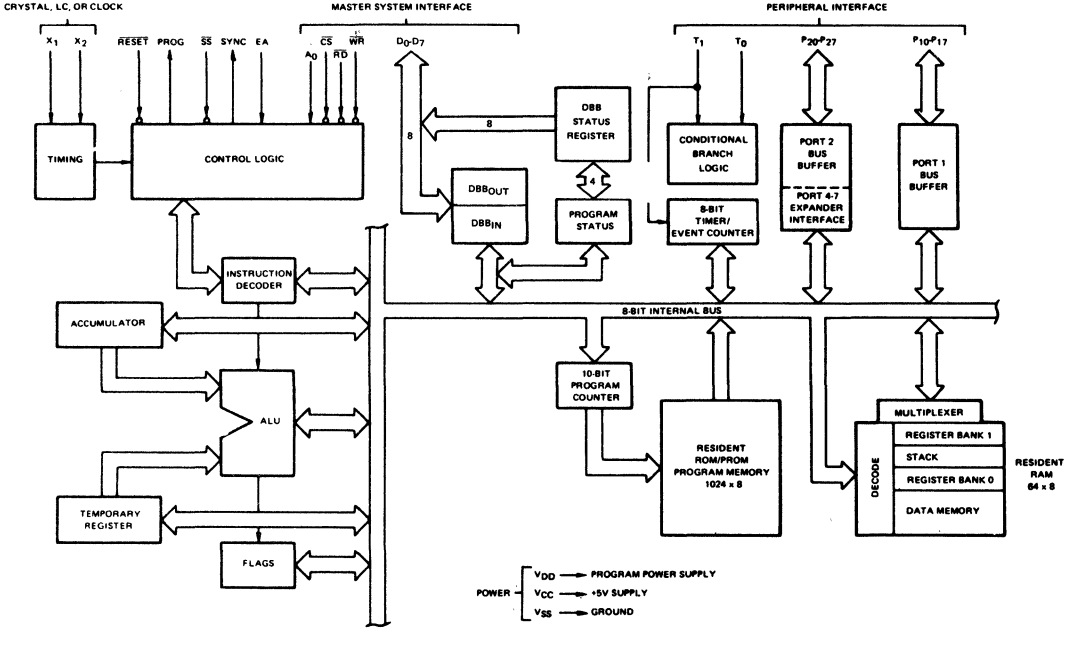
4. P_{24} and P_{25} can be used as either port lines or Buffer Status Flag pins. This feature allows the user to make OBF and IBF status available externally to interrupt the master processor. Upon execution of the EN Flags instruction, P_{24} becomes the OBF pin. When a "1" is written to P_{24}, the OBF pin is enabled and the status of OFB is output. A "0" written to P_{24} disables the OBF pin and the pin remains low. This pin indicates valid data is available from the μPD8041A/8741A. EN Flags instruction execution also enables P_{25} indicate that the μPD8041A/8741A is ready to accept data. A "1" written to P_{25} enables the IBF pin and the status of IBF is available on P25. A "0" written to P_{25} disables the IBF pin.

   EN Flags Instruction Op code – F5H.

5. P_{26} and P_{27} can be used as either port lines or DMA handshake lines to allow DMA interface. The EN DMA instruction enables P_{26} and P_{27} to be used as DRQ (DMA Request) and $\overline{DACK}$ (DMA acknowledge) respectively. When a "1" is written to P_{26}, DRQ is activated and a DMA request is issued. Deactivation of DRQ is accomplished by the execution of the EN DMA instruction, $\overline{DACK}$ anded with $\overline{RD}$, or $\overline{DACK}$ anded with $\overline{WR}$. When EM DMA has been executed, P_{27} ($\overline{DACK}$) functions as a chip select input for the Data Bus Buffer registers during DMA transfers.

   EN DMA Instruction Op Code – E5H.

BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS*

Operating Temperature . . . . . 0°C to +70°C
Storage Temperature (Ceramic Package) . . . . . -65°C to +150°C
Storage Temperature (Plastic Package) . . . . . -65°C to +125°C
Voltage on Any Pin . . . . . -0.5 to +7 Volts ①
Power Dissipation . . . . . 1.5 Watt

COMMENT: Stress above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note: ① With respect to ground.

*T_a = 25°C

DC CHARACTERISTICS

T_a = 0°C to +70°C; V_{DD} = V_{CC} = +5V ± 10%; V_{SS} = 0V

PARAMETER	SYMBOL	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNIT	TEST CONDITIONS
Input Low Voltage (All except X_1 and X_2)	V_{IL}	-0.5		+0.8	V	
Input Low Voltage (X_1 and X_2, $\overline{RESET}$)	V_{IL1}	-0.5		0.6	V	
Input High Voltage (All except X_1, X_2, $\overline{RESET}$)	V_{IH}	2.0		V_{CC}	V	
Input High Voltage (X_1, X_2, $\overline{RESET}$)	V_{IH1}	3.8		V_{CC}	V	
Output Low Voltage (D_0-D_7, SYNC)	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output Low Voltage (All other outputs except PROG)	V_{OL1}			0.45	V	I_{OL} = 1.0 mA
Output Low Voltage (PROG)	V_{OL2}			0.45	V	I_{OL} = 1.0 mA
Output High Voltage (D_0-D_7)	V_{OH}	2.4			V	I_{OH} = -400 μA
Output High Voltage (All other outputs)	V_{OH1}	2.4			V	I_{OH} = -50 μA
Input Leakage Current (T_0, T_1, $\overline{RD}$, $\overline{WR}$, $\overline{CS}$, EA, A_0)	I_{IL}			±10	μA	$V_{SS} \leq V_{IN} \leq V_{CC}$
Output Leakage Current (D_0-D_7; High Z State)	I_{OL}			±10	μA	V_{SS} + 0.45 ≤ $V_{IN} \leq V_{CC}$
V_{DD} Supply Current	I_{DD}			15	mA	
Total Supply Current	$I_{CC} + I_{DD}$			125	mA	
Low Input Source Current (P_{10}-P_{17}; P_{20}-P_{27})	I_{LI}			0.5	mA	V_{IL} = 0.8V
Low Input Source Current (SS; $\overline{RESET}$)	I_{LI1}			0.2	mA	V_{IL} = 0.8V

7

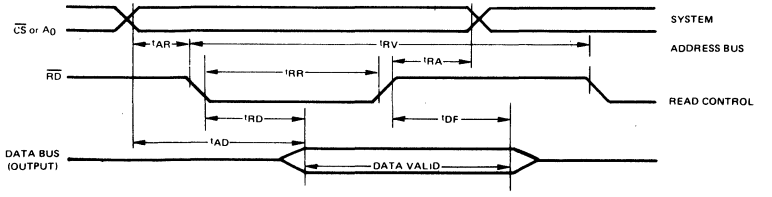
AC CHARACTERISTICS

T_a = 0°C to +70°C; V_{DD} = V_{CC} = +5V ± 10%; V_{SS} = 0V

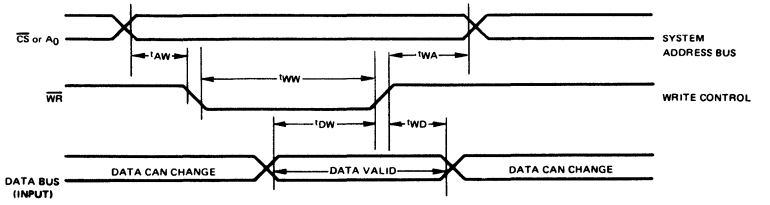
PARAMETER	SYMBOL	LIMITS μPD8041A MIN	LIMITS μPD8041A MAX	LIMITS μPD8741A MIN	LIMITS μPD8741A MAX	UNITS	TEST CONDITIONS
DBB READ							
$\overline{CS}$, A_0 Setup to $\overline{RD}$ ↓	t_{AR}	0		60		ns	
$\overline{CS}$, A_0 Hold after $\overline{RD}$ ↑	t_{RA}	0		30		ns	
$\overline{RD}$ Pulse Width	t_{RR}	250		300	2 x t_{CY}	ns	t_{CY} = 2.5 μs
$\overline{CS}$, A_0 to Data Out Delay	t_{AD}		225		370	ns	C_L = 150 pF
$\overline{RD}$ ↓ to Data Out Delay	t_{RD}		225		200	ns	C_L = 150 pF
$\overline{RD}$ ↑ to Data Float Delay	t_{DF}		100		140	ns	
Cycle Time	t_{CY}	2.5	15	2.5	15	μs	6 MHz Crystal
DBB WRITE							
$\overline{CS}$, A_0 Setup to $\overline{WR}$ ↓	t_{AW}	0		60		ns	
$\overline{CS}$, A_0 Hold after $\overline{WR}$ ↑	t_{WA}	0		30		ns	
$\overline{WR}$ Pulse Width	t_{WW}	250		300	2 x t_{CY}	ns	t_{CY} = 2.5 μs
Data Setup to $\overline{WR}$ ↑	t_{DW}	150		250		ns	
Data Hold after $\overline{WR}$ ↑	t_{WD}	0		30		ns	

TIMING WAVEFORMS

READ OPERATION – DATA BUS BUFFER REGISTER



WRITE OPERATION – DATA BUS BUFFER REGISTER



INSTRUCTION SET

MNEMONIC	FUNCTION	DESCRIPTION	INSTRUCTION CODE D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	CYCLES	BYTES	FLAGS C	AC	F0	F1	IBF	OBF	ST_{4-7}
ACCUMULATOR																			
ADD A, # data	(A) ← (A) + data	Add Immediate the specified Data to the Accumulator.	0 d_7	0 d_6	0 d_5	0 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2	•						
ADD A, Rr	(A) ← (A) + (Rr) for r = 0 7	Add contents of designated register to the Accumulator.	0	1	1	0	1	r	r	r	1	1	•						
ADD A, @ Rr	(A) ← (A) + ((Rr)) for r = 0 1	Add Indirect the contents the data memory location to the Accumulator	0	1	1	0	0	0	0	r	1	1	•						
ADDC A, # data	(A) ← (A) + (C) + data	Add Immediate with carry the specified data to the Accumulator	0 d_7	0 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2	•						
ADDC A, Rr	(A) ← (A) + (C) + (Rr) for r = 0 7	Add with carry the contents of the designated register to the Accumulator	0	1	1	1	1	r	r	r	1	1	•						
ADDC A, @ Rr	(A) ← (A) + (C) + ((Rr)) for r = 0 1	Add Indirect with carry the contents of data memory location to the Accumulator.	0	1	1	1	0	0	0	r	1	1	•						
ANL A, # data	(A) ← (A) AND data	Logical and specified Immediate Data with Accumulator	0 d_7	1 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2							
ANL A, Rr	(A) ← (A) AND (Rr) for r = 0 7	Logical and contents of designated register with Accumulator	0	1	0	1	1	r	r	r	1	1							
ANL A, @ Rr	(A) ← (A) AND ((Rr)) for r = 0 1	Logical and Indirect the contents of data memory with Accumulator	0	1	0	1	0	0	0	r	1	1							
CPL A	(A) ← NOT (A)	Complement the contents of the Accumulator	0	0	1	1	0	1	1	1	1	1							
CLR A	(A) ← 0	CLEAR the contents of the Accumulator	0	0	1	0	0	1	1	1	1	1							
DA A		DECIMAL ADJUST the contents of the Accumulator	0	1	0	1	0	1	1	1	1	1	•						
DEC A	(A) ← (A) 1	DECREMENT by 1 the accumulator's contents	0	0	0	0	0	1	1	1	1	1							
INC A	(A) ← (A) + 1	Increment by 1 the accumulator's contents	0	0	0	1	0	1	1	1	1	1							
ORL A, # data	(A) ← (A) OR data	Logical OR or specified immediate data with Accumulator	0 d_7	1 d_6	0 d_5	0 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2							
ORL A, Rr	(A) ← (A) OR (Rr) for r = 0 7	Logical OR contents of designated register with Accumulator	0	1	0	0	1	r	r	r	1	1							
ORL A, @ Rr	(A) ← (A) OR ((Rr)) for r = 0 1	Logical OR Indirect the contents of data memory location with Accumulator	0	1	0	0	0	0	0	r	1	1							
RL A	(AN + 1) ← (AN) $(A_0) \leftarrow (A_7)$ for N = 0 6	Rotate Accumulator left by 1 bit without carry	1	1	1	0	0	1	1	1	1	1							
RLC A	(AN + 1) ← (AN), N = 0 6 $(A_0) \leftarrow (C)$ $(C) \leftarrow (A_7)$	Rotate Accumulator left by 1 bit through carry	1	1	1	1	0	1	1	1	1	1	•						
RR A	(AN) ← (AN + 1), N = 0 6 $(A_7) \leftarrow (A_0)$	Rotate Accumulator right by 1 bit without carry	0	1	1	1	0	1	1	1	1	1							
RRC A	(AN) ← (AN + 1), N = 0 6 $(A_7) \leftarrow (C)$ $(C) \leftarrow (A_0)$	Rotate Accumulator right by 1 bit through carry	0	1	1	0	0	1	1	1	1	1	•						
SWAP A	$(A_{4-7}) \leftrightarrow (A_{0-3})$	Swap the 2 4-bit nibbles in the Accumulator.	0	1	0	0	0	1	1	1	1	1							
XRL A, # data	(A) ← (A) XOR data	Logical XOR specified immediate data with Accumulator	1 d_7	1 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2							
XRL A, Rr	(A) ← (A) XOR (Rr) for r = 0 – 7	Logical XOR contents of designated register with Accumulator	1	1	0	1	1	r	r	r	1	1							
XRL A, @ Rr	(A) ← (A) XOR ((Rr)) for r = 0 – 1	Logical XOR Indirect the contents of data memory location with Accumulator	1	1	0	1	0	0	0	r	1	1							
BRANCH																			
DJNZ Rr, addr	(Rr) ← (Rr) 1, r = 0 – 7 if (Rr) ≠ 0 (PC 0 7) ← addr	Decrement the specified register and test contents	1 a_7	1 a_6	1 a_5	0 a_4	1 a_3	r a_2	r a_1	r a_0	2	2							
JBb addr	(PC 0 7) ← addr if Bb = 1 (PC) ← (PC) + 2 if Bb = 0	Jump to specified address if Accumulator bit is set.	b_2 a_7	b_1 a_6	b_0 a_5	1 a_4	0 a_3	0 a_2	1 a_1	0 a_0	2	2							
JC addr	(PC 0 – 7) ← addr if C = 1 (PC) ← (PC) + 2 if C = 0	Jump to specified address if carry flag is set	1 a_7	1 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JF0 addr	(PC 0 7) ← addr if F0 = 1 (PC) ← (PC) + 2 if F0 = 0	Jump to specified address if Flag F0 is set.	1 a_7	0 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JF1 addr	(PC 0 7) ← addr if F1 = 1 (PC) ← (PC) + 2 if F1 = 0	Jump to specified address if Flag F1 is set	0 a_7	1 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JMP addr	(PC 8 10) ← addr 8 10 (PC 0 – 7) ← addr 0 7 (PC 11) ← DBF	Direct Jump to specified address within the 2K address block.	a_{10} a_7	a_9 a_6	a_8 a_5	0 a_4	0 a_3	1 a_2	0 a_1	0 a_0	2	2							
JMPP @ A	(PC 0 – 7) ← ((A))	Jump indirect to specified address with with address page.	1	0	1	1	0	0	1	1	2	1							
JNC addr	(PC 0 – 7) ← addr if C = 0 (PC) ← (PC) + 2 if C = 1	Jump to specified address if carry flag is low.	1 a_7	1 a_6	1 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JNIBF addr	(PC 0 – 7) ← addr if IBF = (PC) ← (PC) + 2 if IBF = 1	Jump to specified address if input buffer full flag is low	1 a_7	1 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JOBF	(PC 0 – 7) ← addr if OBF = 1 (PC) ← (PC) + 2 if OBF = 0	Jump to specified address if output buffer full flag is set.	1 a_7	0 a_6	0 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							

7

INSTRUCTION SET (CONT.)

MNEMONIC	FUNCTION		INSTRUCTION CODE D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	CYCLES	BYTES	FLAGS C	AC	F0	F1	IBF	OBF	ST_{4-7}
BRANCH (CONT.)																			
JNT0 addr	(PC 0 7) ← addr if T0 = 0 (PC) ← (PC) + 2 if T0 = 1	Jump to specified address if Test 0 is low	0 a_7	0 a_6	1 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JNT1 addr	(PC 0 7) ← addr if T1 = 0 (PC) ← (PC) + 2 if T1 = 1	Jump to specified address if Test 1 is low	0 a_7	1 a_6	0 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JNZ addr	(PC 0 7) ← addr if A ≠ 0 (PC) ← (PC) + 2 if A = 0	Jump to specified address if accumulator is non-zero	1 a_7	0 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JTF addr	(PC 0 7) ← addr if TF = 1 (PC) ← (PC) + 2 if TF = 0	Jump to specified address if Timer Flag is set to 1	0 a_7	0 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JT0 addr	(PC 0 7) ← addr if T0 = 1 (PC) ← (PC) + 2 if T0 = 0	Jump to specified address if Test 0 is a 1	0 a_7	0 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JT1 addr	(PC 0 7) ← addr if T1 = 1 (PC) ← (PC) + 2 if T1 0	Jump to specified address if Test 1 is a 1	0 a_7	1 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
JZ addr	(PC 0 7) ← addr if A = 0 (PC) ← (PC) + 2 if A ≠ 0	Jump to specified address if Accumulator is 0	1 a_7	1 a_6	0 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2							
CONTROL																			
EN I		Enable the External Interrupt input	0	0	0	0	0	1	0	1	1	1							
DIS I		Disable the External Interrupt input	0	0	0	1	0	1	0	1	1	1							
SEL RB0	(BS) ← 0	Select Bank 0 (locations 0 – 7) of Data Memory.	1	1	0	0	0	1	0	1	1	1							
SEL RB1	(BS) ← 1	Select Bank 0 (locations 24 – 31) of Data Memory	1	1	0	1	0	1	0	1	1	1							
EN DMA		Enable DMA Handshake	1	1	1	1	0	1	0	1	1	1							
EN FLAGS		Enable Interrupt to Master Device	1	1	1	0	0	1	0	1	1	1							
DATA MOVES																			
MOV A, # data	(A) ← data	Move Immediate the specified data into the Accumulator	0 d_7	0 d_6	1 d_5	0 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2							
MOV A, Rr	(A) ← (Rr), r = 0 7	Move the contents of the designated registers into the Accumulator	1	1	1	1	1	r	r	r	1	1							
MOV A, @ Rr	(A) ← ((Rr)), r = 0 1	Move Indirect the contents of data memory location into the Accumulator	1	1	1	1	0	0	0	r	1	1							
MOV A, PSW	(A) ← (PSW)	Move contents of the Program Status Word into the Accumulator	1	1	0	0	0	1	1	1	1	1							
MOV Rr, # data	(Rr) ← data, r = 0 7	Move Immediate the specified data into the designated register	1 d_7	0 d_6	1 d_5	1 d_4	1 d_3	r d_2	r d_1	r d_0	2	2							
MOV Rr, A	(Rr) ← (A), r = 0 7	Move Accumulator Contents into the designated register	1	0	1	0	1	r	r	r	1	1							
MOV @ Rr, A	((Rr)) ← (A), r = 0 1	Move Indirect Accumulator Contents into data memory location	1	0	1	0	0	0	0	r	1	1							
MOV @ Rr, # data	((Rr)) ← data, r = 0 1	Move Immediate the specified data into data memory	1 d_7	0 d_6	1 d_5	1 d_4	0 d_3	0 d_2	0 d_1	r d_0	2	2							
MOV PSW, A	(PSW) ← (A)	Move contents of Accumulator into the program status word	1	1	0	1	0	1	1	1	1	1							
MOVP A, @ A	(PC 0 7) ← (A) (A) ← ((PC))	Move data in the current page into the Accumulator	1	0	1	0	0	0	1	1	2	1							
MOVP3 A, @ A	(PC 0 7) ← (A) (PC 8 10) ← 011 (A) ← ((PC))	Move Program data in Page 3 into the Accumulator	1	1	1	0	0	0	1	1	2	1							
XCH A, Rr	(A) ⇄ (Rr), r = 0 – 7	Exchange the Accumulator and designated register's contents	0	0	1	0	1	r	r	r	1	1							
XCH A, @ Rr	(A) ⇄ ((Rr)), r = 0 1	Exchange Indirect contents of Accumulator and location in data memory	0	0	1	0	0	0	0	r	1	1							
XCHD A, @ Rr	(A 0 – 3) ⇄ ((Rr)) 0 – 3)), r = 0 – 1	Exchange Indirect 4-bit contents of Accumulator and data memory	0	0	1	1	0	0	0	r	1	1							
FLAGS																			
CPL C	(C) ← NOT (C)	Complement Content of carry bit.	1	0	1	0	0	1	1	1	1	1	•						
CPL F0	(F0) ← NOT (F0)	Complement Content of Flag F0	1	0	0	1	0	1	0	1	1	1			•				
CPL F1	(F1) ← NOT (F1)	Complement Content of Flag F1	1	0	1	1	0	1	0	1	1	1				•			
CLR C	(C) ← 0	Clear content of carry bit to 0	1	0	0	1	0	1	1	1	1	1	•						
CLR F0	(F0) ← 0	Clear content of Flag 0 to 0.	1	0	0	0	0	1	0	1	1	1			•				
CLR F1	(F1) ← 0	Clear content of Flag 1 to 0	1	0	1	0	0	1	0	1	1	1				•			
MOV STS, A	ST_4-ST_7 ← A_4-A_7	Move high order 4 bits of Accumulator into status register bits 4-7	1	0	0	1	0	0	0	0	1	1							•

INSTRUCTION SET (CONT.)

MNEMONIC	FUNCTION	DESCRIPTION	INSTRUCTION CODE D7	D6	D5	D4	D3	D2	D1	D0	CYCLES	BYTES	FLAGS C	AC	F0	F1	IBF	OBF	ST_{4-7}
INPUT/OUTPUT																			
ANL Pp, # data	(Pp) ← (Pp) AND data p = 1 2	Logical and Immediate specified data with designated port (1 or 2)	1	0	0	1	1	0	p	p	2	2							
			d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0									
ANLD Pp, A	(Pp) ← (Pp) AND (A 0 3) p = 4 7	Logical and contents of Accumulator with designated port (4 – 7).	1	0	0	1	1	1	p	p	2	1							
IN A, Pp	(A) ← (Pp); p = 1 2	Input data from designated port (1 2) into Accumulator.	0	0	0	0	1	0	p	p	2	1							
IN A, DBB	(A) ← (DBB)	Input strobed DBB data into Accumulator and clear IBF	0	0	1	0	0	0	1	0	1	1					•		
MOVD A, Pp	(A 0 – 3) ← (Pp); p = 4 – 7 (A 4 7) ← 0	Move contents of designated port (4 7) into Accumulator.	0	0	0	0	1	1	p	p	2	1							
MOVD Pp, A	(Pp) ← A 0 – 3; p = 4 7	Move contents of Accumulator to designated port (4 7)	0	0	1	1	1	1	p	p	1	1							
ORLD Pp, A	(Pp) ← (Pp) OR (A 0 3) p = 4 7	Logical or contents of Accumulator with designated port (4 7).	1	0	0	0	1	1	p	p	1	1							
ORL Pp, # data	(Pp) ← (Pp) OR data p = 1 2	Logical or Immediate specified data with designated port (1 2)	1	0	0	0	1	0	p	p	2	2							
			d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0									
OUT DBB, A	(DBB) (A)	Output contents of Accumulator onto DBB and set OBF.	0	0	0	0	0	0	1	0	1	1						•	
OUTL Pp, A	(Pp) ← (A), p = 1 2	Output contents of Accumulator to designated port (1 2).	0	0	1	1	1	0	p	p	1	1							
REGISTERS																			
DEC Rr (Rr)	(Rr) ← (Rr) 1, r = 0 7	Decrement by 1 contents of designated register.	1	1	0	0	1	r	r	r	1	1							
INC Rr	(Rr) ← (Rr) +1; r = 0 7	Increment by 1 contents of designated register	0	0	0	1	1	r	r	r	1	1							
INC @ Rr	((Rr)) ← ((Rr)) + 1, r = 0 1	Increment Indirect by 1 the contents of data memory location.	0	0	0	1	0	0	0	r	1	1							
SUBROUTINE																			
CALL addr	((SP)) ← (PC), (PSW 4 7) (SP) ← (SP) + 1 (PC 8 10) ← addr 8 10 (PC 0 – 7) ← addr 0 7 (PC 11) ← DBF	Call designated Subroutine.	a_{10}	a_9	a_8	1	0	1	0	0	2	2							
			a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0									
RET	(SP) ← (SP) 1 (PC) ← ((SP))	Return from Subroutine without restoring Program Status Word.	1	0	0	0	0	0	1	1	2	1							
RETR	(SP) ← (SP) 1 (PC) ← ((SP)) (PSW 4 7) ← ((SP))	Return from Subroutine restoring Program Status Word.	1	0	0	1	0	0	1	1	2	1							
TIMER/COUNTER																			
EN TCNTI		Enable Internal Interrupt Flag for Timer/Counter output	0	0	1	0	0	1	0	1	1	1							
DIS TCNTI		Disable Internal Interrupt Flag for Timer/Counter output.	0	0	1	1	0	1	0	1	1	1							
MOV A, T	(A) ← (T)	Move contents of Timer/Counter into Accumulator.	0	1	0	0	0	0	1	0	1	1							
MOV T, A	(T) ← (A)	Move contents of Accumulator into Timer/Counter.	0	1	1	0	0	0	1	0	1	1							
STOP TCNT		Stop Count for Event Counter	0	1	1	0	0	1	0	1	1	1							
STRT CNT		Start Count for Event Counter.	0	1	0	0	0	1	0	1	1	1							
STRT T		Start Count for Timer	0	1	0	1	0	1	0	1	1	1							
MISCELLANEOUS																			
NOP		No Operation performed.	0	0	0	0	0	0	0	0	1	1							

Notes
① Instruction Code Designations r and p form the binary representation of the Registers and Ports involved.
② The dot under the appropriate flag bit indicates that its content is subject to change by the instruction it appears in.
③ References to the address and data are specified in bytes 2 and or 1 of the instruction.
④ Numerical Subscripts appearing in the FUNCTION column reference the specific bits affected.

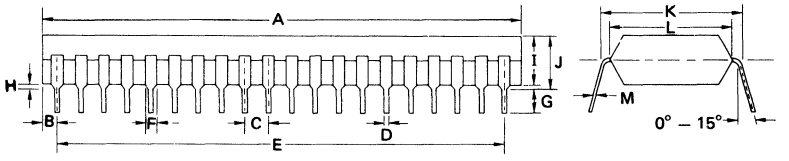
Symbol Definitions:

SYMBOL	DESCRIPTION
A	The Accumulator
AC	The Auxiliary Carry Flag
addr	Program Memory Address (12 bits)
Bb	Bit Designator (b = 0 – 7)
BS	The Bank Switch
BUS	The BUS Port
C	Carry Flag
CLK	Clock Signal
CNT	Event Counter
D	Nibble Designator (4 bits)
data	Number or Expression (8 bits)
DBF	Memory Bank Flip-Flop
F_0, F_1	Flags 0, 1
I	Interrupt
P	"In-Page" Operation Designator
IBF	Input Buffer Full Flag

SYMBOL	DESCRIPTION
P_p	Port Designator (p = 1, 2 or 4 – 7)
PSW	Program Status Word
Rr	Register Designator (r = 0, 1 or 0 – 7)
SP	Stack Pointer
T	Timer
TF	Timer Flag
T_0, T_1	Testable Flags 0, 1
X	External RAM
#	Prefix for Immediate Data
@	Prefix for Indirect Address
$	Program Counter's Current Value
(x)	Contents of External RAM Location
((x))	Contents of Memory Location Addressed by the Contents of External RAM Location.
←	Replaced By
OBF	Output Buffer Full
DBB	Data Bus Buffer

7

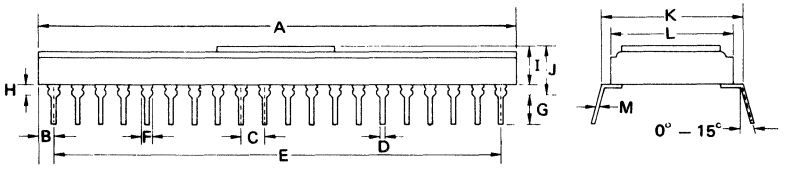
PACKAGE OUTLINE
μPD8041AC
μPD8741AC



(Plastic)

ITEM	MILLIMETERS	INCHES
A	51.5 MAX	2.028 MAX
B	1.62	0.064
C	2.54 ± 0.1	0.10 ± 0.004
D	0.5 ± 0.1	0.019 ± 0.004
E	48.26	1.9
F	1.2 MIN	0.047 MIN
G	2.54 MIN	0.10 MIN
H	0.5 MIN	0.019 MIN
I	5.22 MAX	0.206 MAX
J	5.72 MAX	0.225 MAX
K	15.24	0.600
L	13.2	0.520
M	$0.25^{+0.1}_{-0.05}$	$0.010^{+0.004}_{-0.002}$

μPD8041AD
μPD8741AD



(Ceramic)

ITEM	MILLIMETERS	INCHES
A	51.5 MAX	2.028 MAX
B	1.62	0.064
C	2.54 ± 0.1	0.100 ± 0.004
D	0.50 ± 0.1	0.0197 ± 0.004
E	48.26 ± 0.2	1.900 ± 0.008
F	1.27	0.050
G	3.2 MIN	0.126 MIN
H	1.0 MIN	0.04 MIN
I	4.2 MAX	0.17 MAX
J	5.2 MAX	0.205 MAX
K	15.24 ± 0.1	0.6 ± 0.004
L	$13.5^{+0.2}_{-0.25}$	$0.531^{+0.008}_{-0.010}$
M	0.30 ± 0.1	0.012 ± 0.004